

The FEM approach to the 3D electrodiffusion on 'meshes' optimized with the Metropolis algorithm

I. D. Kosińska

Wroclaw University of Technology

Institute of Biomedical Engineering and Instrumentation

Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland

Abstract

The presented article contains a 3D mesh generation routine optimized with the Metropolis algorithm. The procedure enables to produce meshes of a prescribed volume V_0 of elements. The finite volume meshes are used with the Finite Element approach. The FEM analysis enables to deal with a set of coupled nonlinear differential equations that describes the electrodiffusional problem. Mesh quality and accuracy of FEM solutions are also examined. High quality of FEM type space-dependent approximation and correctness of discrete approximation in time are ensured by finding solutions to the 3D Laplace problem and to the 3D diffusion equation, respectively. Their comparison with analytical solutions confirms accuracy of obtained approximations.

1 Introduction

One from the most important physical processes is electrodiffusion. It describes both diffusional motion of mass and charge flow due to applied electric field. The electric potential distribution is govern by the Poisson equation and total transport of particles is given in terms of the continuity equation [1]. The significance of this equation is broadly described in existing physical, chemical and biological literature [2] and lots of scientific articles, particularly, those which concern properties of nano and micro transport.

Mathematically, equations of electrodiffusion constitute a set of coupled nonlinear equations where the Laplace operator [3, 4, 5] appears together with the first order partial time derivative. The Laplace operator is the basic operator met in many physical situations. Thus the first step to deal with the electrodiffusional problem is to approximate the solution of the Laplace equation with help of the Finite Elements Method. Practically, it means that an appropriate mesh should be designed for a prescribed 3D domain. The mesh must fit well to the physical conditions like e. g. symmetry of the problem. Therefore, different mesh shapes could be desired (spherical, cylindrical, conical or cubic) up to problem. After having accurate basic spatial solutions on appropriate meshes, the problem should be extended to the time-dependent case of the diffusion equation by finding discrete approximation in time. It could be done by means of truncated Taylor series or other single-step procedures like the Crank–Nicolson scheme [6, 7] or the Gurtin’s approach to finite element approximation in terms of variational principle [8]. From now, further extension of above-presented computations involving non-linear terms could be easily implemented and numerically solved using the Newton’s method [9].

2 Equation of electrodiffusion

The equation of electrodiffusion [1] has the form

$$\begin{aligned}\nabla \cdot \mathbf{J}_i + \frac{\partial n_i}{\partial t} &= 0 \\ \mathbf{J}_i &= -\tilde{D}_i \left(\nabla n_i + \frac{z_i e n_i}{kT} \nabla \phi \right)\end{aligned}\tag{1}$$

where n_i is a number of i -th ions, ϕ - electric potential, \tilde{D}_i - the diffusion coefficient of i -th particles, k_B - Boltzmann constant, T - temperature, z_i - valence of the i -th kind of ions, e - electric charge. To find the electric potential ϕ the Poisson equation must be solved

$$\nabla^T(\epsilon_0 \epsilon \nabla \phi) + \rho = 0\tag{2}$$

where $\rho = \sum_i \rho_i$, $\rho_i = z_i e n_i$. Thus equations (1) and (2) both constitute the system of coupled equations.

2.1 FEM approach

Next, they can be solved numerically using the Finite Element Method [6] where the problem is represented as

$$\begin{aligned}\int_{\Omega} \mathbf{v}^T \mathbf{A}(\mathbf{u}) d\Omega &\equiv \int_{\Omega} [v_1 A_1(\mathbf{u}) + v_2 A_2(\mathbf{u}) + v_3 A_3(\mathbf{u})] d\Omega = 0 \\ \int_{\Gamma} \tilde{\mathbf{v}}^T \mathbf{B}(\mathbf{u}) d\Gamma &\equiv \int_{\Gamma} [\tilde{v}_1 B_1(\mathbf{u}) + \tilde{v}_2 B_2(\mathbf{u}) + \tilde{v}_3 B_3(\mathbf{u})] d\Gamma = 0\end{aligned}\tag{3}$$

where

$$\mathbf{u} = \begin{bmatrix} n_+ \\ n_- \\ \phi \end{bmatrix} \quad (4)$$

and \mathbf{v} and $\tilde{\mathbf{v}}$ are sets of arbitrary functions equal in number to the number of equations (or components of \mathbf{u}) involved. $A_1(\mathbf{u})$, $A_2(\mathbf{u})$ and $A_3(\mathbf{u})$ are given by the following formulas, respectively

$$\begin{aligned} A_1(\mathbf{u}) &= \nabla^T \left(-k_+ \mathbf{u}^T \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \nabla \mathbf{u}^T \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} - \tilde{D}_+ \nabla \mathbf{u}^T \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \right) + [1, 0, 0] \frac{\partial}{\partial t} \mathbf{u} \\ A_2(\mathbf{u}) &= \nabla^T \left(-k_- \mathbf{u}^T \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \nabla \mathbf{u}^T \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} - \tilde{D}_- \nabla \mathbf{u}^T \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \right) + [0, 1, 0] \frac{\partial}{\partial t} \mathbf{u} \\ A_3(\mathbf{u}) &= \epsilon_0 \epsilon \nabla^T \nabla \mathbf{u}^T \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + [1, -1, 0] z e \mathbf{u}. \end{aligned} \quad (5)$$

where $k_i = \frac{D_i z_i e}{k_B T}$, $i = \{+, -\}$. An expression $\mathbf{B}(\mathbf{u})$ gives the boundary conditions on Γ , however, we choose a *forced* type of boundary conditions on Γ i. e.

$$\begin{aligned} \phi - \phi_{boun} &= 0 \\ n_i - n_{i, boun} &= 0. \end{aligned} \quad (6)$$

Let us substitute $\mathbf{u} \approx \sum_a N_a \mathbf{I} \begin{bmatrix} \tilde{n}_+^a \\ \tilde{n}_-^a \\ \tilde{\phi}^a \end{bmatrix} = \mathbf{N} \tilde{\mathbf{u}}$ and put $\mathbf{v} = \sum_b w_b \mathbf{I} \delta \tilde{\mathbf{u}}_b$ where $w_b = N_b$. In that way, we end up with the Galerkin formulation of the problem.

2.1.1 Discrete approximation in time

In turn, we can approximate the nodal electric potential

$$\phi(x, y, z, t) = \sum_a N_a(x, y, z) \tilde{\phi}^a(t) \quad (7)$$

and number of particles

$$n_i(x, y, z, t) = \sum_a N_a(x, y, z) \tilde{n}_i^a(t) \quad (8)$$

at a time t_n by

$$\tilde{\phi}^a(t_n) \approx \tilde{\phi}_n^a \quad (9)$$

$$\tilde{n}_i^a(t_n) \approx \tilde{n}_{i,n}^a \quad (10)$$

and taking advantage from an evaluation of $\tilde{\mathbf{u}}_n = \begin{bmatrix} \tilde{n}_{+,n} \\ \tilde{n}_{-,n} \\ \tilde{\phi}_n \end{bmatrix}$ in the Taylor series we obtain

$$\tilde{\mathbf{u}}_n \approx \tilde{\mathbf{u}}_{n-1} + \Delta t \dot{\tilde{\mathbf{u}}}_{n-1} + \beta (\Delta t)^2 \ddot{\tilde{\mathbf{u}}}_{n-1} + O(\Delta t^3) \quad (11)$$

where β takes values from $[0, 1]$ and Δt denotes time step. After incorporating it into a general form of time-dependent equations $A_{\{1,2\}}(\mathbf{u})$

$$K(\mathbf{u}) + \mathbf{C}\dot{\mathbf{u}} = 0 \quad (12)$$

where $K(\mathbf{u})$ represents these parts of $A_{\{1,2\}}(\mathbf{u})$ with a space-dependent operator we get time approximation for a given node

$$K(N_a \mathbf{I} \mathbf{u}_n^a) + \mathbf{C} N_a \mathbf{I} \left\{ \frac{1}{\beta \Delta t} (\mathbf{u}_n^a - \mathbf{u}_{n-1}^a) - \frac{1-\beta}{\beta} \dot{\mathbf{u}}_{n-1}^a \right\} = 0. \quad (13)$$

When $\beta = 1$ an approximate solution to the semi-discrete equations at each time t_n is given by the Euler „backward” scheme

$$K(\mathbf{N}\tilde{\mathbf{u}}) + \mathbf{C}\mathbf{N} \frac{1}{\Delta t} \tilde{\mathbf{u}}_n = \mathbf{C}\mathbf{N} \frac{1}{\Delta t} \tilde{\mathbf{u}}_{n-1} \quad (14)$$

otherwise, the expression for u_n^a is as follows

$$\mathbf{C} N_a \mathbf{I} \frac{1}{\beta \Delta t} \mathbf{u}_n^a + K(N_a \mathbf{I} \mathbf{u}_n^a) - \mathbf{C} N_a \mathbf{I} \frac{1}{\beta \Delta t} \mathbf{u}_{n-1}^a + \frac{1-\beta}{\beta} K(N_a \mathbf{I} \mathbf{u}_{n-1}^a) = 0. \quad (15)$$

2.1.2 Space-dependent term

After integration $NK(\mathbf{N}\tilde{\mathbf{u}})$ by parts, one obtains a mixed set of linear and nonlinear equations

$$\begin{aligned} & \int_{\Omega} (\nabla N_b)^T k_+ \left(\sum_a N_a \tilde{n}^{+,a} \right) \nabla \left(\sum_c N_c \tilde{\phi}^c \right) d\Omega + \int_{\Omega} (\nabla N_b)^T \tilde{D}_+ \nabla \left(\sum_a N_a \tilde{n}^{+,a} \right) d\Omega \\ & + \frac{1}{\Delta t} \int_{\Omega} N^b \sum_a N_a (\tilde{n}_n^{+,a} - \tilde{n}_{n-1}^{+,a}) d\Omega = 0 \\ & \int_{\Omega} (\nabla N_b)^T k_- \left(\sum_a N_a \tilde{n}^{-,a} \right) \nabla \left(\sum_c N_c \tilde{\phi}^c \right) d\Omega + \int_{\Omega} (\nabla N_b)^T \tilde{D}_- \nabla \left(\sum_a N_a \tilde{n}^{-,a} \right) d\Omega \\ & + \frac{1}{\Delta t} \int_{\Omega} N^b \sum_a N_a (\tilde{n}_n^{-,a} - \tilde{n}_{n-1}^{-,a}) d\Omega = 0 \\ & - \int_{\Omega} (\nabla N_b)^T \epsilon_0 \epsilon \nabla \left(\sum_a N_a \tilde{\phi}^a \right) d\Omega + z_+ e \int_{\Omega} N^b \left(\sum_a N_a \tilde{n}^{+,a} \right) d\Omega \\ & + z_- e \int_{\Omega} N^b \left(\sum_a N_a \tilde{n}^{-,a} \right) d\Omega = 0 \\ & b = 1, \dots, M \end{aligned} \quad (16)$$

and a corresponding set of boundary terms for ϕ and n^i where $i = \{+, -\}$

$$\begin{aligned} -k_i \oint_{\Gamma} N^b \frac{\partial}{\partial n} \left\{ \left(\sum_a N_a \tilde{n}^{i,a} \right) \nabla \left(\sum_c N_c \tilde{\phi}^c \right) \right\} d\Gamma - \tilde{D}_i \oint_{\Gamma} N^b \frac{\partial}{\partial n} \left\{ \nabla \left(\sum_a N_a \tilde{n}^{i,a} \right) \right\} d\Gamma \\ + \epsilon \epsilon_0 \oint_{\Gamma} N^b \frac{\partial}{\partial n} \left\{ \nabla \left(\sum_a N_a \tilde{\phi}^a \right) \right\} d\Gamma = 0 \quad b = 1, \dots, M \end{aligned} \quad (17)$$

where $\frac{\partial}{\partial n}$ denotes derivative normal to Γ . Presented-above spatially temporal discretization has been done for the case with $\beta = 1$ at each node. If the *forced* boundary conditions (see Eq. (6)) are imposed on Γ_ϕ and Γ_n , respectively, then all terms in Eq. (17) can be neglected by restricting the choice of N_b functions to those which equal 0 on Γ . Let's denote integrals from Eq. (16) as

$$\begin{aligned} K_{ac}^b &= \int_{\Omega} (\nabla N_b)^T N_a \nabla N_c d\Omega = \sum_e \int_{\Omega^e} (\nabla N_b)^T N_a \nabla N_c d\Omega^e = \sum_e K_{ac}^{b,e} \\ \tilde{K}_a^b &= \int_{\Omega} (\nabla N_b)^T \nabla N_a d\Omega = \sum_e \int_{\Omega^e} (\nabla N_b)^T \nabla N_a d\Omega^e = \sum_e \tilde{K}_a^{b,e} \\ K_a^b &= \int_{\Omega} N_a N^b d\Omega = \sum_e \int_{\Omega^e} N_a N^b d\Omega^e = \sum_e K_a^{b,e}. \end{aligned} \quad (18)$$

where \sum_e with $e = 1, \dots, E$ denotes sum over elements. In 3D space tetrahedral elements seem to be a natural choice of finite volume elements. Then indices a, b, c take four values each (an element has four nodes) from the set of $1, \dots, M$ values.

2.2 Tetrahedral elements

For tetrahedral linear elements shape functions N_i can be assumed as equal area coordinates L_i given by the formula

$$L_i = \frac{a_i + b_i x + c_i y + d_i z}{6V^e}, \quad i = 1, 2, 3, 4 \quad (19)$$

where V^e is a volume of tetrahedron. The following integration formula can be useful

$$\int_{\Omega^e} L_1^\alpha L_2^\beta L_3^\mu L_4^\nu dx dy dz = \frac{\alpha! \beta! \mu! \nu!}{(\alpha + \beta + \mu + \nu + 3)!} 6V^e \quad (20)$$

in order to calculate integrals $K_{ac}^{b,e}$, $\tilde{K}_a^{b,e}$ and $K_a^{b,e}$. Shape functions for linear elements are $N_a = L_a$ for $a = 1, 2, 3, 4$. This gives

$$\begin{aligned}
K_{a,c}^{b,e} &= \int_{\Omega^e} (\nabla L^b)^T L_a \nabla L_c d\Omega^e = \frac{1}{4!6V^e} [b^b, c^b, d^b] [b_c, c_c, d_c]^T = \\
&\quad \frac{1}{4!6V^e} (b^b b_c + c^b c_c + d^b d_c) \\
\tilde{K}_a^{b,e} &= \int_{\Omega^e} (\nabla L^b)^T \nabla L_a d\Omega^e = \frac{1}{36V^e} (b^b b_a + c^b c_a + d^b d_a) \\
K_a^{b,e} &= \int_{\Omega^e} L_a L^b d\Omega^e = \frac{6V^e}{5!} \quad \text{when } a \neq b \\
K_a^{a,e} &= \int_{\Omega^e} L_a L^a d\Omega^e = \frac{12V^e}{5!}.
\end{aligned} \tag{21}$$

Finally, we have

$$\begin{aligned}
&\left\{ k_+ \left(\sum_e K_{ac}^{b,e} \right) \tilde{\phi}^c + \tilde{D}_+ \sum_e \tilde{K}_a^{b,e} + \frac{1}{\Delta t} \sum_e K_a^{b,e} \right\} \tilde{n}^{+,a} = f^{+,b}, \\
&\left\{ k_- \left(\sum_e K_{ac}^{b,e} \right) \tilde{\phi}^c + \tilde{D}_- \sum_e \tilde{K}_a^{b,e} + \frac{1}{\Delta t} \sum_e K_a^{b,e} \right\} \tilde{n}^{-,a} = f^{-,b}, \\
&\epsilon_0 \epsilon \tilde{\phi}^a \sum_e \tilde{K}_a^{b,e} = |z_i| e \sum_e K_a^{b,e} (\tilde{n}^{+,a} - \tilde{n}^{-,a}) \\
&\quad \text{where } a, b, c = 1, \dots, M
\end{aligned} \tag{22}$$

where $f^{i,b} = \frac{1}{\Delta t} \left(\sum_e K_a^{b,e} \right) \tilde{n}_{n-1}^{i,a}$ with $i = +, -$ and only these nodes a, b and c that participate in the particular element e can give a non-zero contribution to the sums of the general type $\sum_e K^e$.

Let's assume that all values of \tilde{n}^a are known at a time t_n . Then we can solve the third equation in (22) obtaining the result

$$\tilde{\phi}^a = \frac{|z_i| e}{\epsilon_0 \epsilon} \left\{ \sum_e \tilde{K}_a^{b,e} \right\}^{-1} \left(\sum_e K_a^{b,e} \right) (\tilde{n}^{+,a} - \tilde{n}^{-,a}) \tag{23}$$

where $\left\{ \sum_e \tilde{K}_a^{b,e} \right\}^{-1}$ denotes elements of the inverse matrix to \tilde{K} . After substitution the solution for $\tilde{\phi}$ to Eq. (22) we get

$$\begin{aligned}
&\tilde{n}^{i,a} \left\{ \frac{k_i z e}{\epsilon_0 \epsilon} \left\{ \sum_e \tilde{K}_c^{b,e} \right\}^{-1} \left(\sum_e K_c^{b,e} \right) (\tilde{n}^{+,c} - \tilde{n}^{-,c}) \left(\sum_e K_{ac}^{b,e} \right) \right. \\
&\quad \left. + \tilde{D}_i \sum_e \tilde{K}_a^{b,e} + \frac{1}{\Delta t} \sum_e K_a^{b,e} \right\} - f^{i,b} = 0.
\end{aligned} \tag{24}$$

where $i = +, -$.

2.2.1 Newton's method

The above written set of equations is of a nonlinear type. Let's denote all of them as

$$F(\tilde{n}^1, \dots, \tilde{n}^M) = 0 \quad (25)$$

Thus to solve it we have to employ the iterative Newton's method [9]. It means that we have to start from an initial guess of $\{\tilde{n}_0^i\}_{i=1}^M$ values. And during next iterations for $k = 0, 1, \dots$

$$\tilde{n}_{k+1} = \tilde{n}_k - \left\{ F'(\tilde{n}_k^1, \dots, \tilde{n}_k^M) \right\}^{-1} F(\tilde{n}_k^1, \dots, \tilde{n}_k^M) \quad (26)$$

the solution should be achieved. $F'(\tilde{n}_k^1, \dots, \tilde{n}_k^M)$ denotes the following matrix of partial derivatives:

$$F'(\tilde{n}_k^1, \dots, \tilde{n}_k^M) = \begin{bmatrix} \frac{\partial F^1}{\partial n_k^1} & \cdots & \frac{\partial F^1}{\partial n_k^M} \\ \vdots & \dots & \vdots \\ \frac{\partial F^M}{\partial n_k^1} & \cdots & \frac{\partial F^M}{\partial n_k^M} \end{bmatrix} \quad (27)$$

where

$$\begin{aligned} \frac{\partial F^b}{\partial n_k^a} &= \left\{ \frac{kze}{\epsilon_0 \epsilon} \left\{ \sum_e \tilde{K}_c^{b,e} \right\}^{-1} \sum_e K_c^{b,e} \tilde{n}_k^c \sum_e K_{ac}^{b,e} + \tilde{D} \sum_e \tilde{K}_a^{b,e} \right. \\ &\quad \left. + \frac{1}{\Delta t} \sum_e K_a^{b,e} \right\} + \frac{2kze}{\epsilon_0 \epsilon} \left\{ \sum_e \tilde{K}_c^{b,e} \right\}^{-1} \sum_e K_c^{b,e} \tilde{n}_k^a \sum_e K_{ac}^{b,e} \end{aligned} \quad (28)$$

where $c \neq a$.

3 Diffusion equation

Putting $k = 0$ in the equation of electrodiffusion we neglect the electrostatic term. It leads to the following equation describing diffusion in \mathfrak{R}^n [4, 5]

$$\begin{cases} u_t - D\Delta u = 0 & \text{in } \mathfrak{R}^n \times (0, \infty), \\ u = g & \text{on } \mathfrak{R}^n \times \{t = 0\}, \end{cases} \quad (29)$$

where D denotes a diffusion coefficient. This kind of equation represents an initial value problem. Assuming that the considered domain Ω in \mathfrak{R}^3 is of a cubic type $[x_{min}, x_{max}] \times [y_{min}, y_{max}] \times [z_{min}, z_{max}]$ let us take $u = 0$ as a boundary condition. Now we seek a solution of the equation (29) which satisfies this boundary condition and prescribed initial condition at the time $t = 0$. The solution of the equation is approximated by the triple sum

$$u(x, y, z, t) = \sum_{k_x=1}^{\infty} \sum_{k_y=1}^{\infty} \sum_{k_z=1}^{\infty} v_{0,k_x,k_y,k_z} e^{-(k_x^2 + k_y^2 + k_z^2)Dt} \sin(k_x x) \sin(k_y y) \sin(k_z z), \quad (30)$$

where v_{0,k_x,k_y,k_z} are unknown coefficients that must be determinated from the initial condition:

$$g = u(\cdot, 0) = \sum_{k_x=1}^{\infty} \sum_{k_y=1}^{\infty} \sum_{k_z=1}^{\infty} v_{0,k_x,k_y,k_z} \sin(k_x x) \sin(k_y y) \sin(k_z z). \quad (31)$$

In the case of the domain being $[0, \pi] \times [0, \pi] \times [0, \pi]$ and $g = \text{const}$ the solution has the form

$$u(x, y, z, t) = \frac{64g}{\pi^3} \sum_{k_x, k_y, k_z=1,3,\dots}^{\infty} \frac{1}{k_x k_y k_z} e^{-(k_x^2 + k_y^2 + k_z^2)Dt} \sin(k_x x) \sin(k_y y) \sin(k_z z). \quad (32)$$

In the case of cylindrical domain defined by $r \in [0, r_0]$, $\theta \in [0, 2\pi)$ and $z \in [0, \pi]$, and with the boundary condition of the form $u = 0$ the solution of Eq. (29) can be expanded in an absolutely and uniformly convergent series of the form

$$u(r, \theta, z, t) = \sum_{n=0}^{\infty} \sum_{m, k_z=1}^{\infty} J_n(k_{n,m} r / r_0) (a_{n,m} \cos(n\theta) + b_{n,m} \sin(n\theta)) \sin(k_z z) e^{-((k_{n,m}/r_0)^2 + k_z^2)Dt} \quad (33)$$

where $k_{n,m}$ are the zeros of the Bessel functions and $a_{n,m}, b_{n,m}$ are constants that must be found from the initial condition by making use of the *orthogonality relation* for the trigonometric and Bessel functions.

4 Laplace equation

On the other hand, assuming that the time derivative in the diffusion equation (29) equals 0 and putting $D = 1$ we end up with the boundary value problem of the Laplace type [4, 5]

$$\begin{cases} \Delta u = 0 & \text{in } \Omega, \\ u = g & \text{on } \partial\Omega. \end{cases} \quad (34)$$

Let's consider Ω being a cubic domain i. e. $[0, \pi] \times [0, \pi] \times [0, \pi]$. And for the g function equals 0 everything on $\partial\Omega$ apart from $g(x = \pi, y, z) = \phi_0$ one can approximate the exact solution by

$$\phi(x, y, z) = \frac{16\phi_0}{\pi^2} \sum_{n,m=1,3,\dots}^{\infty} \frac{\sinh(\sqrt{n^2 + m^2}x) \sin(ny) \sin(mz)}{nm \sinh(\sqrt{n^2 + m^2}\pi)} \quad (35)$$

For $\phi(x, y, z) = v(r)$ where $r = (x^2 + y^2 + z^2)^{1/2}$ the Laplace equation has the solution defined in \mathbb{R}^3 for $r \neq 0$

$$\phi(x, y, z) = \frac{1}{3\alpha(3)} \frac{1}{(x^2 + y^2 + z^2)^{1/2}} \quad (36)$$

where $\alpha(3)$ denotes volume of $B(0,1)$ in \mathbb{R}^3 and equals $\frac{\pi^{3/2}}{\Gamma(3/2 + 1)}$.

5 Three-dimensional mesh generation

Below are listed a few technical remarks referring to the mesh generation routine applied to obtain a designed 3D mesh.

5.1 Initial mesh

An initial mesh is built on the basis of main surface nodes (*outer* nodes) which define a figure's shape. The whole figure is considered as divided into perpendicular to z -axis layers. Thus the *outer* nodes are distributed on the edges of layers. In the center of each layer and also in the middle between two layers are located inner nodes. They are connected with *outer* nodes creating in this way the main figure's construction. Initial mesh elements obtained in such a manner are of tetrahedral shape.

5.2 Figure's surface

The boundary of the figure is defined by set of *surface* equations for vertical and horizontal segment lines linking *outer* nodes. After each mesh iteration new nodes are created and labeled as *outer* or *inner* ones according to surface equations. Moreover, the location of each node (i. e. on which exactly vertical, horizontal line or surface patch the node is lying) is also stored.

5.3 New elements creation

New elements are created by a division of already existing elements. At the beginning of the routine, the surface of division mainly connects a new node born on the longest element edge with two other nodes belonging to that mesh element and one node from the divided edge. The procedure constitutes a 3D extension of the 2D mesh generation routine described already in [14]. However, during the routine a number of small elements is increasing, and the division of the longest bar is not anymore the optimal way of proceeding. That is why, before choosing an edge to the division volume of elements common to it is checked. The edge that will not produce new elements having its volume smaller than an assumed **critical volume** is chosen to be cut.

5.4 Mesh optimization – Metropolis algorithm

The optimization is done with help of the Metropolis algorithm. The system energy is calculated as a sum of discrepancies between an element volume V^e and assumed element volume $V_0 = h_0^3\sqrt{2}/12$ where h_0 denotes a prescribed length of the edge

$$E = \sum_e (V^e - V_0)^2. \quad (37)$$

Thus the smaller a degeneracy from a designed volume distribution the more optimal state. The Metropolis routine starts from a nodal configuration given by described above procedure. The main point is to reach the optimal global configuration by ascertaining local optimal states. They arise from such a configuration of i -th node and its neighboring nodes which gives smaller energy E^i . This partial energy is calculated from the sum Eq. (37) taken over elements containing the node of interest. To compute new positions for each node (giving new configuration) the following expression is put forward

$$\mathbf{p}_{i,new} = \mathbf{p}_i - k_s \sum_{ij} \delta \mathbf{r}_{ij} \quad (38)$$

$$\delta \mathbf{r}_{ij} = \sum_j (|\mathbf{p}_i - \mathbf{p}_j| - h_0) \frac{\mathbf{p}_i - \mathbf{p}_j}{|\mathbf{p}_i - \mathbf{p}_j|} \quad (39)$$

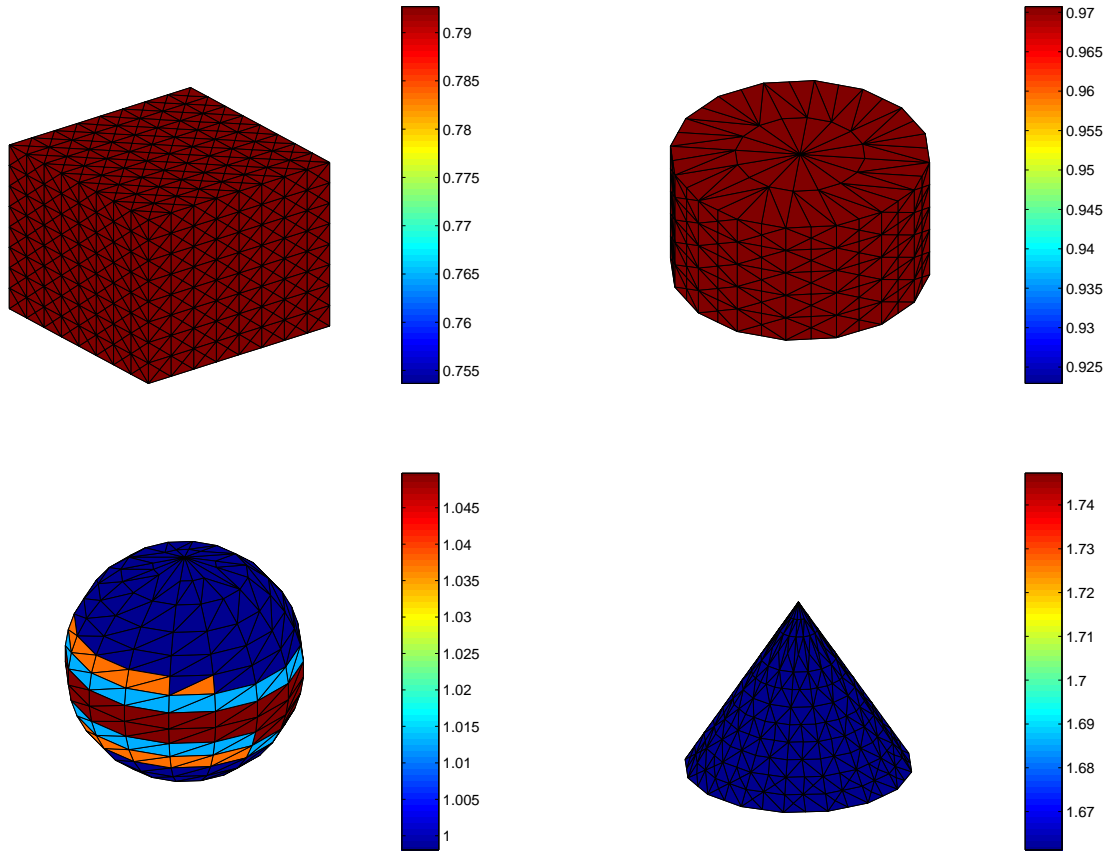


Figure 1: The figure presents four regular shapes (cube, cylinder, sphere and cone) obtained with the non-optimized routine. Colorbars show variations in final elements volume.

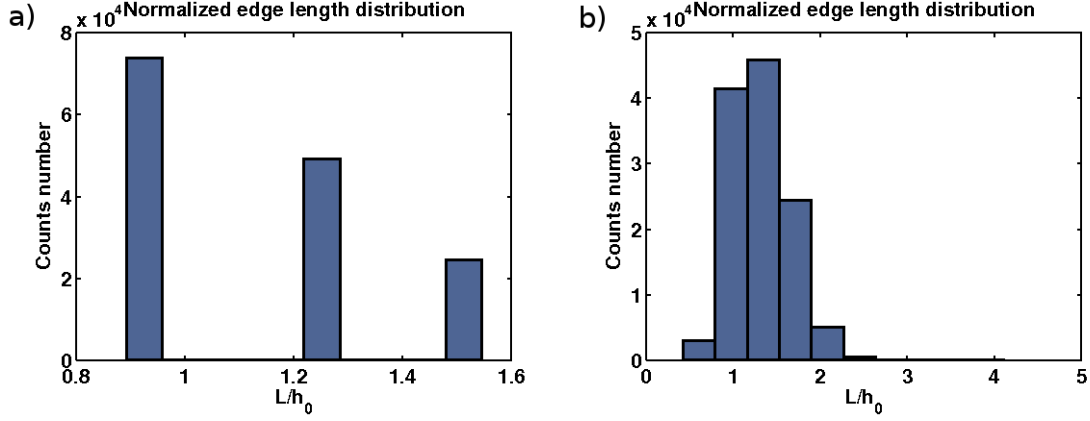


Figure 2: The figure presents a distribution of a normalized mesh edge length L/h_0 for meshes in a cubic domain $[0, \pi] \times [0, \pi] \times [0, \pi]$ with a) unique elements created for $h_0 = 0.22$ and b) meshes optimized with the Metropolis algorithm with $h_0 = 0.2$.

where k_s denotes a shifting strength and $\mathbf{p}_i - \mathbf{p}_j$ is the length of ij edge. The value of k_s determines the strength of a nodal shift and varies from 0 to 1. It is also worthy considering to choose its value as a random number from uniform distribution $U(0, 1)$. Within the Metropolis routine the transition probability is calculated by the formula

$$P(p_i \rightarrow p_{i,new}) = e^{-\Delta E_i/k_B T} \quad (40)$$

where k_B is a Boltzmann constant (here set as 1), T temperature and $\Delta E_i = E_{i,new} - E_i$ is a difference between energies of these two states. If a value of P is greater than a random number from $U(0, 1)$ new state is accepted. Otherwise the old one is preserved. All above-described local Metropolis steps can lead to different global configurations. Therefore, for each division number this distribution of nodes which gives a lower energy of the total system should be kept. To find this, again the Metropolis rule is employed, but this time changes in the total energy of the whole system are examined. To estimate maximal temperature T_{max} (in expression (40)) a range of changes in potential energy corresponding to current number of elements must be found. Moreover, in each global Metropolis step temperature might decrease according to $T = \eta T$ where parameter $\eta < 1$.

5.5 Delaunay reconfiguration routine

To improve mesh quality the following transformations are applied [6].

- Three elements common to an edge are transformed to two elements, when one of the elements fails to satisfy the Delaunay criterion [10].
- Four elements common to an edge are transformed to a new configuration of four elements, when one of the elements does not meet the Delaunay criterion [10]. The new pattern is chosen from two different possibilities [6].

Additionally, too small boundary elements could be destructed by a projection of its internal node to the center of the outer patch of element that is opposite to it. Such approach is justified in the case of boundary elements, however, in the case of internal ones leads to creation of so-called irregular nodes.

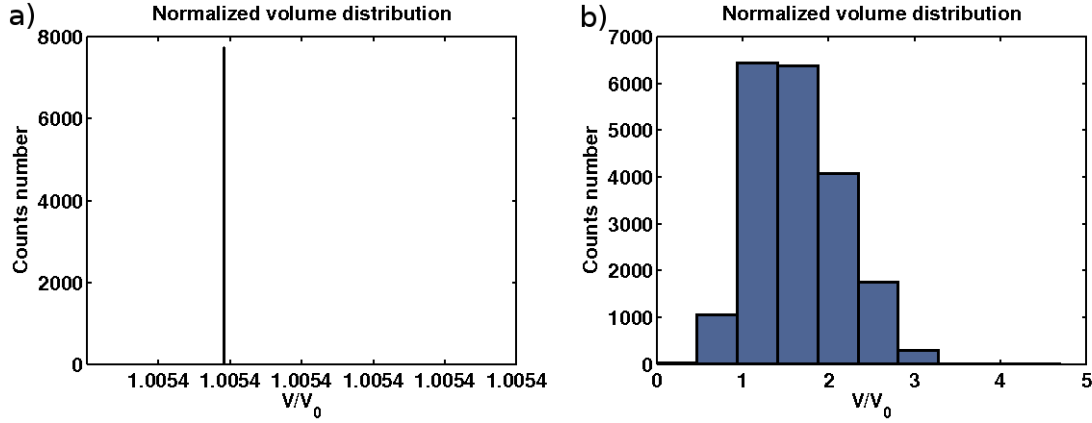


Figure 3: The picture presents a distribution of a normalized mesh element volume V/V_0 for the same domain as in Fig. 2 and for a) unique elements; b) elements with $V_0 = 9 \times 10^{-4}$ optimized with the Metropolis algorithm.

6 Results

6.1 The Laplace equation

The accuracy of FEM approximation of the Laplace equation on different meshes were examined. Numerical results vs. analytical ones for cubic and spherical domain are presented in Fig. (4). The relative difference between both analytical and numerical solutions has been calculated as

$$(\phi_{num} - \phi_{anal}) / \max(\phi_{anal}). \quad (41)$$

The Laplace equation has been solved for the cubic domain $[0, \pi] \times [0, \pi] \times [0, \pi]$ with potential function $\phi = 0$ everywhere on the boundary Γ apart from one its side at $x = \pi$ where potential $\phi = 1$ and for the spherical domain with the boundary conditions imposed by putting an elementary charge outside the sphere in $[0, 0, 2\pi]$. The exact solutions for both considered cases are evaluated precisely in Sec. 4. FEM approximation has been computed for the „linear” order of tetrahedron [6]. However, the comparison between both orders of approximation i.e. „linear” and „quadratic” for the uniform mesh with $V_{elem} = 0.0202$ has been performed. The formulas for higher orders of approximation i.e. quadratic and cubic can be found in [6]. The results show that mean discrepancy between numerical and analytical solutions calculated according to Eq. (41) for the Laplace equation equal -0.0061 ± 0.0153 in the linear case and 0.0004 ± 0.0082 in quadratic approximation, respectively. Thus in further studies linear approximation will be used as being sufficiently accurate. In the case of cubic domain a mesh of unique element volume (non-optimized) has been applied in contrast to the spherical domain where mesh has been used after its enhancement with both the Metropolis algorithm and the Delaunay routine.

6.2 The diffusion equation

To test accuracy of discrete approximation in time the equation of diffusion (see Sec. 3) has been examined. To find $P(x, y, z, t)$ particles distribution at t_n moment in time the Taylor expansion (see Eq. 11) with $\beta = 0$ has been applied. The diffusion equation with

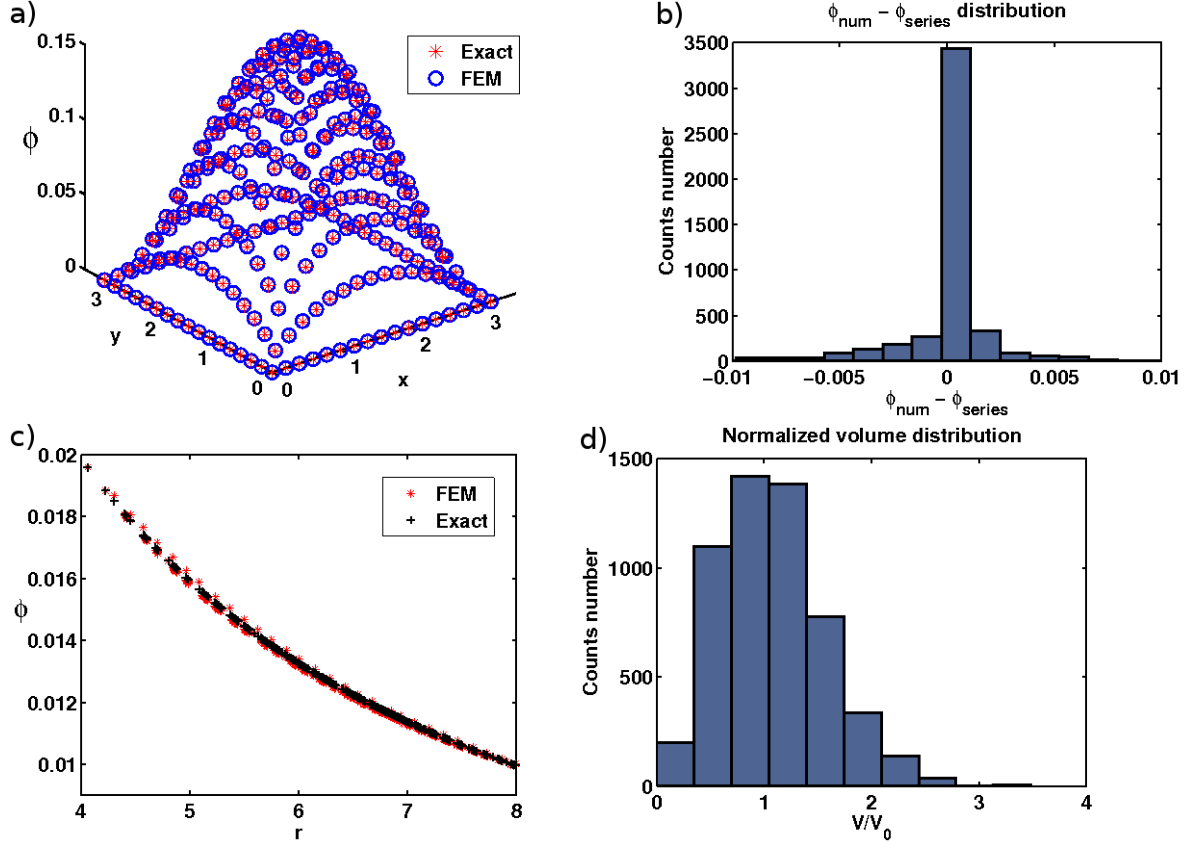


Figure 4: The picture shows a) FEM approximation (linear) of the Laplace problem at the center of the cubic domain i. e. at $z = \pi/2$ (as described in Sec. 4) vs. the analytical result together with b) a distribution of differences between numerical and analytical solutions obtained for each node in Ω domain; c) FEM approximation (linear) vs. the exact solution of the Laplace equation for the spherical domain with the boundary values defined by putting an elementary charge outside the sphere in $[0, 0, 2\pi]$ b) the volume distribution in the spherical domain optimized with the Metropolis algorithm with elements of a prescribed volume V_0 equals 0.0075.

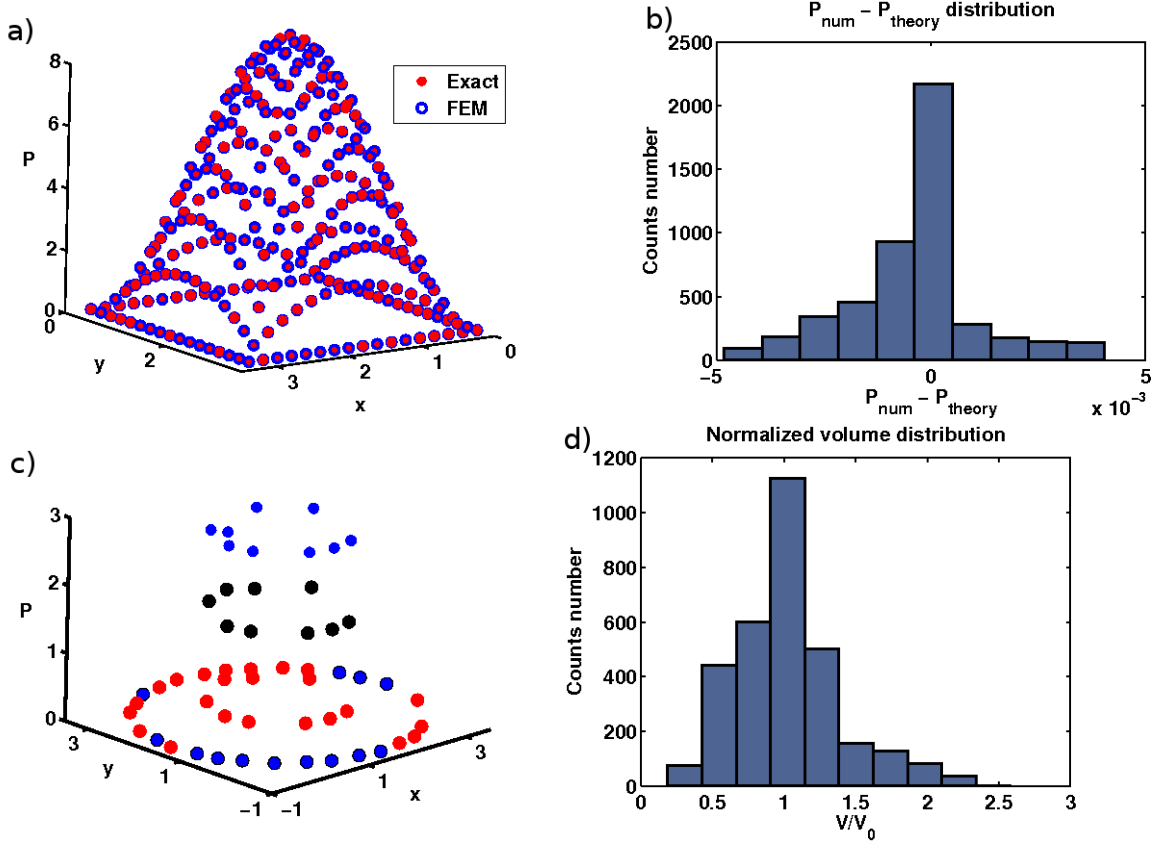


Figure 5: The picture shows a) FEM approximation (linear) of the initial-boundary value problem in the center of the cubic domain i. e. at $z = \pi/2$ and at the time $t = 0.19$ with $\Delta t = 0.01$ [units] vs. the analytical result together with b) a distribution of differences (Eq. 41) between numerical and analytical solutions obtained for each node in Ω domain; c) FEM approximation (linear) of the diffusion equation for the cylindrical domain at $z = \pi/2$ and at the following times: $t_0 = 0$ (blue), $t_{mid} = 0.5$ (black) and $t_{end} = 1.0$ (red) with $\Delta t = 0.01$ [units]; d) volume profile of elements within the cylindrical domain where $V_0 = 0.015$ and $h_0 = 0.5$; mesh quality were enhanced with help of the Metropolis algorithm.

the following initial condition $g(\cdot, 0) = -x(x - \pi)y(y - \pi)z(z - \pi)$ for the cubic domain and $g(\cdot, 0) = |(r - R_0)z(z - \pi)|$ where R_0 denotes a radius of the cylindrical domain has been solved. The boundary value of the $P(x, y, z, t)$ is set as 0. Results for both domains: cubic (with uniform elements – see Fig. 3a) and cylindrical (see Fig. 1), this one tuned to the designed element volume with the Metropolis recipe, are shown in Fig. 5. In the case of cylindrical domain mean values of the ratio $P(\vec{x}, t_i)/P(\vec{x}, t_{i+10})$ calculated at each point of domain for times $i = 0, 10, \dots, 100$ have the average value 1.2699 ± 0.0049 .

6.3 The electrodiffusion equation

The system of coupled equations describing process of electrodiffusion (1) written in terms of the FEM method (see Eq. (22)) with the following values of constants: $k_+ = k_- = 0.05$, $D_+ = D_- = 0.05$ and the time step equals 0.01 has been numerically solved by using the Newton's algorithm. The boundary values of n_+ , n_- , ϕ are set as 1 at $x = 0$, $x = \pi$, $y = 0$, $y = \pi$, and $z = 0$ and equal 2 at $z = \pi$. An initial guess of n_+ , n_- and ϕ

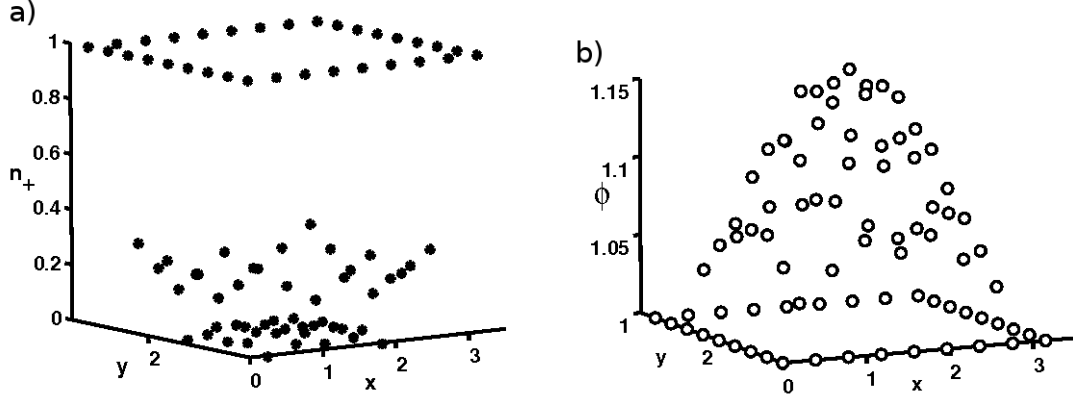


Figure 6: The picture shows FEM approximation (linear) of the system of electrodiffusion equations being the boundary value problem. The solutions are depicted in the center of the cubic domain i. e. at $z = \pi/2$ and at the time $T = 0.39$ with the values of parameters $\Delta t = 0.01$ [units], $k_+ = k_- = 0.05$, $D_+ = D_- = 0.05$ [units] a) the distribution of cations n_+ ; b) the profile of the potential ϕ ; computations have been performed on the uniform mesh with the volume of tetrahedron $V_0 = 0.01$ and the element size $h_0 = 0.44$.

distributions has been chosen as 0 everywhere in the domain apart from its boundaries. The system of equations has been computed up to the final time $T = 0.39$. Fig. 6 presents obtained profiles of cations n_+ and the potential ϕ at the center of the domain i. e. at $z = \pi/2$. There is no visible difference between cations n_+ and anions n_- distributions so the latter is not shown. The maximum of $n_{+,i+1} - n_{+,i}$ is decaying from 0.023 for $i = 0$ ($t = 0$) to 0.0093 for $i = 37$ ($t = 0.37$). The maximum of difference between n_+ and n_- distributions computed at each node equals $1.3e - 09$. Employing physical notion it means that the system of charged particles is electroneutral. Moreover, the system of particles is tending to its stationary state.

Additionally, components j_x, j_y and j_z of the total flux of n_+ particles flowing through the domain Ω have been computed. They are shown in Fig. 7. Presence of a difference in an amount of n_+ particles at the both sides of Ω in the z -direction i. e. at $z = 0$ and $z = \pi$ causes a non-zero flow along z axis whereas a lack of such a difference in the two other directions i. e. x and y leads to the vanishing flows j_x and j_y in the center of the domain.

7 Conclusions

The presented software offers a 3D mesh generation routine as well as its further application to the 3D electrodiffusional problem.

The proposed mesh generator offers a confident way to create a quite uniform mesh built with elements having desired volume. Mesh elements have been adjusted to assumed sizes by making use of both the Metropolis algorithm and the Delaunay criterion. Mesh quality depicted in histograms occurs to be fairly satisfactory. Moreover, goodness of obtained meshes together with robustness of their applications to the Finite Element Method have been also tested by solving the 3D Laplace problem and the 3D diffusion equation on them. Comparison between these numerical solutions and analytical results shows very good agreement.

To find solutions to a nonlinear problem defined by a system of coupled equations de-

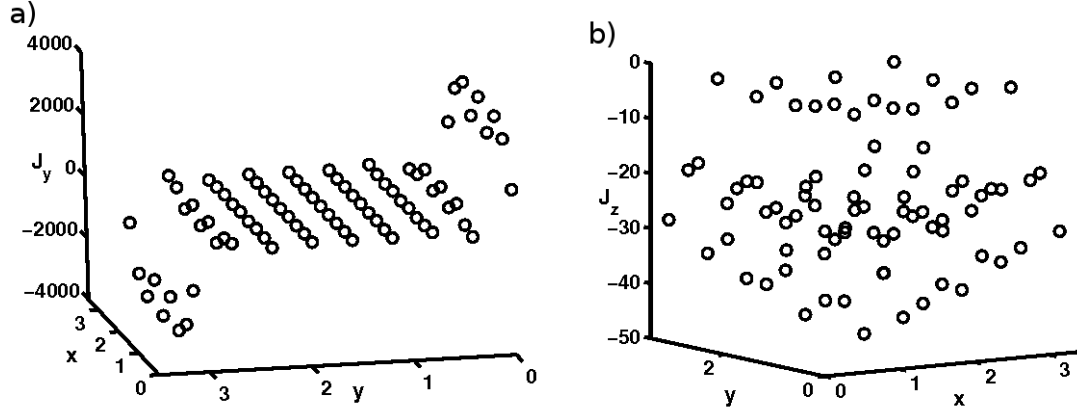


Figure 7: The picture presents FEM approximation (linear) to the flux components obtained for the electrodiffusional problem. The solutions are shown at the center of the cubic domain i. e. at $z = \pi/2$ and at the time $T = 0.39$. Computations have been done with the following values of parameters: the time step $\Delta t = 0.01$ [units], nonlinear term multipliers $k_+ = k_- = 0.05$ [units], diffusion coefficients $D_+ = D_- = 0.05$ [units]. This figure shows approximated solutions for flux components for cations n_+ in the y direction a) and in the z direction b). The computations have been performed on the uniform mesh with the volume of tetrahedron $V_0 = 0.01$ and the element size $h_0 = 0.44$.

scribing electrodiffusion the FEM approach and the Newton method have been jointly applied. Analysis of obtained results confirms usefulness of the presented solver to deal with nonlinear differential problems.

References

- [1] M. Toda and R. Kubo, *Statistical Physics II. Nonequilibrium Statistical Mechanics* (Springer-Verlag, 1985)
- [2] N. G. van Kampen, *Stochastic Processes in Physics and Chemistry* (North-Holland, Amsterdam, 1981); E. Nelson, *Dynamical Theories of Brownian Motion* (University Press, Princeton, NY, 1967);
- [3] R. P. Feynman, R. B. Leighton, M. Sands, *The Feynman Lectures on Physics* (Addison Wesley Publishing Company, 2005)
- [4] L. C. Evans, *Partial Differential Equations* (American Mathematical Society, 1998)
- [5] R. Courant and D. Hilbert, *Methods of mathematical physics* vol. 1 (Interscience Publishers Ltd., London, 1953)
- [6] O. C. Zienkiewicz, R. L. Taylor and J. Z. Zhu, *The Finite Element Method. Its Basis & Fundamentals* (6-th eds., Elsevier Ltd., 2005)
- [7] J. Crank and P. Nicolson, *A practical method for numerical integration of solutions of partial differential equations of heat conduction type.*, Proc. Camb. Phil. Soc., **43**, 50 (1947)

- [8] M. E. Gurtin, Q. Appl. Math., **22**, 252-256 (1964); M. E. Gurtin, Arch. Rat. Mech. Anal., **16**, 34-50 (1969); E. L. Wilson and R. E. Nickell, Nucl. Eng. Design, **4**, 1-11 (1966)
- [9] C. T. Kelley, *Solving Nonlinear Equations with Newton's Method in Fundamentals of Algorithms* SIAM, 2003; J. Brzózka, L. Dorobczyński, *MATLAB. Środowisko obliczeń naukowo-technicznych*. (PWN, Warszawa, 2008)
- [10] B. Delaunay, Sur la sphère vide, Izvestia Akademii Nauk SSSR, Otdelenie Matematicheskikh i Estestvennykh Nauk **7**, 793 (1934)
- [11] N. Metropolis, S. Ulam, J. Amer. Stat. Assoc. **44**(247), 335 (1949)
- [12] The source code of Octave is freely distributed GNU project, for more info please go to the following web page <http://www.gnu.org/software/octave/>.
- [13] MATLAB, www.mathworks.com, 2010.
Licence: Wrocław Centre for Networking and Supercomputing
(<http://www.wcss.wroc.pl/english/sc.php>)
- [14] I. D. Kosińska, The FEM approach to the 2D Poisson equation in 'meshes' optimized with the Metropolis algorithm, arXiv:1008.2715; also submitted to „Programming and Computer Software”;